FINANCIAL MARKET NEWS-SENTIMENT ANALYSIS

This is a data of financial top 25 news for the day and task is to train and predict model for overall sentiment analysis

Import Library

import pandas as pd

import numpy as np

Import Dataset

df = pd.read_csv(r'https://raw.githubusercontent.com/YBI-Foundation/Dataset/main/Financial%20Market%20News.csv',encoding = "ISO-8859-1")

df.head()

	Date	Label	News 1	News 2	News 3	News 4	News 5	News 6	News
0	01- 01- 2010	0	McIlroy's men catch cold from Gudjonsson	Obituary: Brian Walsh	Workplace blues leave employers in the red	Classical review: Rattle	Dance review: Merce Cunningham	Genetic tests to be used in setting premiums	Ope review: Bohè
1	02- 01- 2010	0	Warning from history points to crash	Investors flee to dollar haven	Banks and tobacco in favour	Review: Llama Farmers	War jitters lead to sell- off	Your not- so-secret history	Revie T Northe Sinfo
2	03- 01- 2010	0	Comment: Why Israel's peaceniks feel betrayed	Court deals blow to seizure of drug assets	An ideal target for spooks	World steps between two sides intent on war	What the region's papers say	Comment: Fear and rage in Palestine	Pove a resentme fue Palestini
3	04- 01- 2010	1	£750,000- a-goal Weah aims parting shot	Newcastle pay for Fletcher years	Brown sent to the stands for Scotland qualifier	Tourists wary of breaking new ground	Canary Wharf climbs into the FTSE 100	Review: Bill Bailey	Revie Classi
4	05- 01- 2010	1	Leeds arrive in Turkey to the silence of the fans	One woman's vision offers loan lifeline	Working Lives: How world leaders worked	Working Lives: Tricks of the trade	Working Lives: six- hour days, long lunches and	Pop review: We Love UK	Wo mu: revie Mari Mor

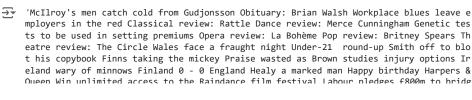
df.info()

```
News 3
                     4101 non-null
                                       object
                     4101 non-null
                                       object
           News 4
      6
           News 5
                     4101 non-null
                                       object
                     4101 non-null
           News 6
                                       object
           News 7
                     4101 non-null
                                       object
                     4101 non-null
      9
           News 8
                                       object
      10
         News 9
                     4101 non-null
                                       object
      11 News 10 4101 non-null
                                       object
          News 11
                     4101 non-null
      12
                                       object
      13
           News 12
                     4101 non-null
                                       object
      14 News 13
                     4101 non-null
                                       object
                     4101 non-null
      15
          News 14
                                       object
      16 News 15 4101 non-null
                                       object
          News 16
                     4101 non-null
                                       object
      18
          News 17
                     4101 non-null
                                       object
      19 News 18 4101 non-null
                                      object
      20 News 19 4101 non-null
                                       object
          News 20
                     4101 non-null
                                       object
      22 News 21
                     4101 non-null
                                      object
      23 News 22 4101 non-null
                                       object
      24 News 23
                    4100 non-null
                                       object
      25 News 24 4098 non-null
                                      obiect
      26 News 25 4098 non-null
                                       object
     dtypes: int64(1), object(26)
     memory usage: 865.2+ KB
df.shape
→ (4101, 27)
df.columns

→ Index(['Date', 'Label', 'News 1', 'News 2', 'News 3', 'News 4', 'News 5',
             'News 6', 'News 7', 'News 8', 'News 9', 'News 10', 'News 11', 'News 12', 'News 13', 'News 14', 'News 15', 'News 16', 'News 17', 'News 18', 'News 19', 'News 20', 'News 21', 'News 22', 'News 23', 'News 24',
             'News 25'],
            dtype='object')
```

Get Feature Selection

```
' '.join(str(x) for x in df.iloc[1,2:27])
→ 'Warning from history points to crash Investors flee to dollar haven Banks and tobacco
     in favour Review: Llama Farmers War jitters lead to sell-off Your not-so-secret history
     Review: The Northern Sinfonia Review: Hysteria Review: The Guardsman Opera: The Marriag
     e of Figaro Review: The Turk in Italy Deutsche spells out its plans for diversification
     Traders' panic sends oil prices skyward TV sport chief leaves home over romance Leader:
     Hi-tech twitch Why Wenger will stick to his Gunners Out of luck England hit rock bottom
     Wilkinson out of his denth Kinsella snarks Trish nower nlav Rrown hanished as Scots reh
df.index
RangeIndex(start=0, stop=4101, step=1)
len(df.index)
→ 4101
news = []
for row in range(0,len(df.index)):
    news.append(' '.join(str(x) for x in df.iloc[row,2:27]))
type(news)
→ list
news[0]
```



```
X = news
type(X)

→ list
```

Get Feature Text Conversion to Bag of Words

Get Train Test Split

```
from sklearn.model_selection import train_test_split
X_{\text{train}}, X_{\text{test}}, y_{\text{train}}, y_{\text{test}} = \text{train\_test\_split}(X, y, \text{test\_size} = 0.3, \text{ stratify} = y, \text{random\_state} = 2529)
from sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier(n_estimators=200)
rf.fit(X_train,y_train)
                 {\tt RandomForestClassifier}
      RandomForestClassifier(n_estimators=200)
y_pred = rf.predict(X_test)
from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
confusion_matrix(y_test,y_pred)
→ array([[161, 420],
             [180, 470]])
print(classification_report(y_test,y_pred))
<del>_</del>
                     precision
                                     recall f1-score
                                                          support
                            0.47
                                                  0.35
                                                               581
                           0.53
                                       0.72
                                                  0.61
                                                               650
```

accuracy			0.51	1231
macro avg	0.50	0.50	0.48	1231
weighted avg	0.50	0.51	0.49	1231

Start coding or generate with AI.